DOCUMENT RESUME

ED 398 923 IR 055 981

AUTHOR McCollum, Roy
TITLE A Pretty Good Paper about Pretty Good Privacy.
PUB DATE 95
NOTE 11p.; In: The Internet--Flames, Firewalls and the Future. Proceedings for the 1995 Conference of the Council for Higher Education Computing Services (CHECS) (Roswell, New Mexico, November 8-10, 1995).
PUB TYPE Guides - Non-Classroom Use (055) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Access to Information; *Coding; Computer Networks; Computers; *Computer Security; Computer Software; *Computer Software Evaluation; *Electronic Mail; Information Dissemination; Information Retrieval; Information Transfer; *Privacy; Users (Information)
IDENTIFIERS Data Protection; Data Security; Instructions; *Key Encryption

ABSTRACT

With today's growth in the use of electronic information systems for e-mail, data development and research, and the relative ease of access to such resources, protecting one's data and correspondence has become a great concern. "Pretty Good Privacy" (PGP), an encryption program developed by Phil Zimmermann, may be the software tool that will provide a person with a secure method to keep mail, manuscripts, and data private. PGP uses a two-key method of encryption. With PGP, a person gives out their public key to all who might send them encrypted messages. The person's private key, which they do not divulge, is then the only key that can access the encrypted messages, so information is secure. This paper contains detailed installation instructions, basic features and strengths of using PGP for e-mail purposes, and information on where and how to obtain current versions of the public domain PGP software. (Author/SWC)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 398 923

A PRETTY GOOD PAPER ABOUT PRETTY GOOD PRIVACY

by: Roy McCollum, MIS at NMSVH

With today's growth in the use of electronic information systems for E-mail, data development and research, the relatively easy access to such resources by all, protecting your data and correspondence has become a great concern. Every time you use E-mail, you're giving system administrators, postmasters, and many others the opportunity to read your data. What is needed is a secure method that lets you keep private your mail, your manuscripts and your data. Pretty Good Privacy (PGP), an encryption program developed by Phil Zimmermann may be just the software tool you need.

In this presentation I would like to demonstrate the strengths of PGP for E-mail purposes in its simple form. This presentation will not encompass all the possibilities that this software can or might possibly do. I will also discuss where and how one can obtain the current versions of the public domain PGP.

With PGP, you can encrypt messages that can be read only by a person using a special decryption key. This keeps your private communication with others secure and accessible.

Throughout history keeping information private for whatever reason has had many methods and encryption was difficult. Many times you were required to give the same secret code (key) to each person communicated with. However, any individual that held the knowledge of the encryption code (key) could decrypt and read communication whether or not they were the intended recipients, not a very secure system. Probably one of the most famous encryption methods, in recent history, is the Enigma system developed by the Nazis during WWII.

With PGP, no one can decrypt your file except the person you present it to, provided you encrypted your file with that person's public key. PGP uses a two-key method of encryption: a private key that only you have and a public key that you freely give to others.

On the surface, this system doesn't sound very secure or manageable. Why freely give out a key to all who might present you with encrypted information? However, with PGP, a person uses your public key to send you encrypted information. With PGP's method only your private key can decrypt this information, and as you might imagine, you never give your private key to anyone. With PGP's two-key encryption system, your information is secure.

At this point we have a general concept of how PGP might work for us. During this presentation we will explore how PGP can secure your file. But first let's look at how the PGP software is installed. The programs that make up PGP are available for several computer operating systems such as DOS/Windows, Macintosh, UNIX, OS/2, VAX and VMS. In this presentation, I'll discuss the DOS installation procedures since many people have access to this particular operating system, keeping in mind that it is similar to the UNIX installation procedures. Depending on your operating system, you may have to compile the source code that is made available by Phil Zimmermann.

Before you can install PGP on your system, you'll need to make sure you have a copy of the program. The current version is PGP 2.6.2. It generally comes

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
William L. Adkins
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR055981

BEST COPY AVAILABLE

in the ZIP compressed-format, so you will need to have a copy of PKUNZIP (ver.2.04g) on your computer in order to decompress PGP into its various files.

We will create the following directory on our local or home directory:

drive:\pgp262

For convenience sake we will locate the file PGP262.ZIP in the \pgp262 directory and use the PKUNZIP program to decompress the file with the following command line:

pkunzip pgp262.zip (Assuming PKUNZIP is found in the PATH variable.)

PGP262.ZIP is a double-nested zip file. After performing the above command you will see the following file list:

PGP262.ZIP (Original file.)
PGP262I.ASC (Signature file.)
SETUP.DOC (Current setup documentation.)
PGP262I.ZIP (Programs, license, documentation.)

It is best to read the SETUP.DOC file for any current information concerning the changes to this installation process.

Now we will decompress the programs, license and documentation files and use the -d switch with PKUNZIP. This will create the proper subdirectory for the documentation files. If it is not used, the documentation will be dropped into the current directory.

pkunzip -d pgp262i.zip (Assuming PKUNZIP is found in the PATH variable.)

After successfully decompressing the compressed files, your directory list should look similar to the following illustration:

Note: The files size, date and times are omitted for commenting.

Directory of C:\PGP262
Volume in drive C is MS-DOS_6
Volume Serial Number is 1EF1-4B29

.	<DIR>		(Current directory)
..	<DIR>		(Parent directory)
CONFIG	TXT		(The PGP configuration file)
DOC		<DIR>	(Directory holding documentation)
ES	HLP		(Spanish help file)
FR	HLP		(French help file)
KEYS		ASC	(Public keys of PGP developers)
LANGUAGE	TXT		(Selected prompt lines by selected language)
MITLICEN	TXT		(License Agreement from MIT)

PGP	**EXE**	**(The Pretty Good Privacy program)**
PGP	**HLP**	**(PGP help file data)**
PGP262	**ZIP**	**(Nested Compressed files hold all files)**
PGP262I	**ASC**	**(Confirmation signature file)**
PGP262I	**ZIP**	**(Compressed file hold files without asterisk)**
README	**DOC**	**(Current information about the enclosed files)**
RSALICEN	**TXT**	**(RSA license agreement)**
SETUP	**DOC**	**(Information on the setup of PGP)**

[other directory information]

The \PGP262\DOC directory will or should include text-file manuals with information on PGP. You can refer to these text files if you want to learn more about PGP or you might pick up Simson Garfinkel's book titled, PGP, published by O'Reilly & Associates. Simson's book will give you a bit of the history in Phil Zimmerman's development of PGP and some depth in the components used in the PGP programs.

Having now made the PGP program files available for our use we must modify some of our DOS environment settings. This is generally done with a simple ASCII text editor like DOS's EDIT or Window's Notepad. The DOS file that we will alter is AUTOEXEC.BAT. This file is found in the root directory (i.e. C:\>). (Using word processors such as WordPerfect or Microsoft Word and saving the modified file in their default format will cause errors from within the AUTOEXEC.BAT file.)

Whenever you start to modify boot-process files, it is always wise to make a copy of the original file. In our case we will copy the AUTOEXEC.BAT file to AUTOEXEC.ORG. That way, if something goes wrong, we can restore the original AUTOEXEC.BAT with its commands. When making a backup, use the following command:

copy c:\autoexec.bat c:*.org

There are a couple of global-environment variables that need to be defined and an addition to the PATH environment variable. If you have a well-written AUTOEXEC.BAT file, you should find these variables defined near the top of the file.

At this point we will open our text editor and load AUTOEXEC.BAT. First we will append the PGP262 directory context to the PATH statement. The following should be somewhat representative of a modified PATH statement's text:

path=c:\;c:\dos;c:\windows;c:\pgp262

Next we will add two environment variables at some location within the AUTOEXEC.BAT file. Generally you should find all your global-environment variables located in one area of the file. The first one that we will add is the

PGPATH variable. This DOS global-variable is used by PGP to locate its support files and should be defined to point to PGP's context. Its syntax should be on a line by itself and entered as follows:

set pgpath=c:\pgp262

Finally, we will set the time-zone variable to represent the local time-zone in which the program will be used or that should be reflected by PGP. Enter the following syntax:

set tz=xxxnxxx

Replace xxxnxxx with text that reflects your desired time zone (e.g., EST5EDT for Eastern, CST6CDT for Central, MST7MDT for Mountain and PST8PDT for Pacific.) Our syntax should look like the following and also
be on a line by itself: SET TZ=MST7MDT. And if you're anything like me you will have to do this twice because you incorrectly typed one of the commands thus causing a syntax error when I reboot the computing machine to activate the changes in the AUTOEXEC.BAT file.

Having now successfully booted our computer, we need to create the private and public keys. Let me say that again: Before encryption of a file can take place with PGP, you must make a private key and a public key. The private key is for you only and should never be given to the public in any way, shape or form.

To begin the generation of our private key in this presentation we need to change to our PGP directory and at the DOS prompt enter: PGP -kg. You will see the screen output and program prompt as shown below:

C:\PGP262> pgp -kg
Pretty Good Privacy<tm> 2.6.2 - Public-key encryption for the masses.
1990-1994 Philip Zimmermann, Phil's Pretty Good Software. 11 Oct 94
Uses the RSAREF<tm> Toolkit, which is copyright RSA Data Security, Inc.
Distributed by the Massachusetts Institute of Technology.
Export of this software may be restricted by the U.S. government.
Current time: 1995/10/28 12:35 GMT
Pick your RSA key size:

1) **512 bits - Low commercial grade, fast but less secure**
2) **768 bits - High commercial grade, medium speed, good security**
3) **1024 bits - "Military" grade, slow, highest security**

Choose 1, 2, or 3, or enter desired number of bits:_

PGP is now prompting for a key size and the greater the key size, the more secure PGP's encryption will be. By using the military-grade key size, we will have the best encryption and still have optimum speed on today's average microcomputer. Upon entering 3 at the prompt, PGP will then prompt us for our user ID. PGP recommends that you use your name followed by your E-mail address, such as:

H.G. Wells <HG_Wells@novels.com>

PGP then requires a pass phrase. This phrase is used to unlock the private key, which enables us to decrypt messages. Use a phrase that's easy to remember and type, don't use anything short and simple, such as your significant other's name, your pet Fluffy's name, or your Social-Security (not so secure) number. In our presentation we will use the pass phrase of:

THE UNDYING FIRE AND PHILOSOPHICAL AND THEOLOGICAL SPECULATIONS.

It's important to remember your pass phrase because you'll need to enter the exact phrase each time you want to unlock your private key. PGP is case, space and punctuation sensitive so pay attention to what and how you type your pass phrase. To PGP, "PHILOSOPHICAL" is not the same as "Philosophical". And remember, NEVER write down your pass phrase. Most security violations occur when someone finds your pass phrase written down.

Upon entering our pass phrase at the prompt and pressing the Enter key, PGP will ask us to enter our pass phrase a second time and check for a confirmed match. PGP will then prompt us to enter a number of random keys. PGP will be using the time between keys pressed for random numbers, so don't hit the same key over and over at the same rhythm. Type in phrases from your childhood (no the clean ones), your favorite hangout's menu or you might type the "99 Bottles of Beer on the Wall" song (you know the one you were trying to sing in the bar last night.) Type until PGP beeps and at this point PGP will display a wait message as it generates the keys.

When the program completes the key generation we will be beeped again and we will notice two new files in the PGP directory: PUBRING.PGP and SECRING.PGP. The file PUBRING.PGP is our public key-ring and it contains our public key and will hold the public keys of other people to whom you'll encrypt information files. The private key that we use to decrypt files with is in the SECRING.PGP file (one thing about PGP's key rings - they will not wear a hole in your pocket.)

Now, at this point we would want to copy both of the ring files to a blank floppy disk and keep it in a safe place. What is a safe place? NOT next to the computer where someone could find the disk and open the files. Remember, they are useless without your pass phrase, so don't write down your pass phrase and place it with the disks. Also keep in mind that diskettes are subject to magnetic fields and will not last forever.

Next, we need to create a text file that will contain our public key. Persons wanting to send you encrypted files will use this public key. At the DOS prompt, in our case, enter:

pgp -kxa H.G. Wells

The PGP directory will now include a file called WELLS.ASC and the text in this file is YOUR public key. Your public key information will look something like this:

--BEGIN PGP PUBLIC KEY BLOCK--
Version: 2.6.2

Adsfasdfl-9904r, czo9809qerokueru]wet] m]oet01b]5\- 2]-]4][obv\\[5yeiy
\rt\uyen oyuoe5o5o[u9]5-0ty8i09eruvqtvubqwkfbasraebPaoirt9vbuqerito
pvwerutvur9tuvwetivtasfasdfjkuiop=]4368741dffafasdfasdfasdfdghjfgjh
vbcubb68ol8yoiuybtr4f29g8ynjogrcfXFTEET8HJRSSERTWQETGYDFGD
=DRdt
--END PGP PUBLIC KEY BLOCK--

Now we have our public key to give to our associates. Your friends will be able to send encrypted files to you and, since they also have PGP, you can send encrypted files in return. Here in our presentation, we will send a file to our friend AnnVeronica@prison.gov.

But first, we'll need Ann's public key. She has sent her public key to us in an E-mail message that she saved in a cake, I mean file named VERONICA.ASC. At this point we can add Ann's public key from this file to our public key ring by using the following command:

pgp -ka VERONICA.ASC

Whenever we add a new key to our public-keyring file, PGP will ask if you can certify that the key is genuine (truly the one created by the recipient.) Certification is a subject we will not go into detail with at this time, so we will answer No to PGP's prompt. Certification is a method in which you can attest to the validity of a public key's owner.

Next, we are going to write a letter to Ann and in this message we will include our public key so that she will be able to send us encrypted mail later. We save our letter to a file called ANNLTR. Now, using PGP, we will encrypt this file by using the command:

pgp -seat ANNLTR Ann -u H.G.

OK, let us take a closer at this command, it's switches and parameters. The format of the encrypt command is:

pgp -seat filename recipient's_userid -u my_userid

In our example encrypt command, -seat and -u are PGP commands, as defined:

- **s Sign the file.**
- **e Encrypt the file.**
- **a Use ASCII characters only.**
- **t Keep line feeds with returns.**
- **u User's secret key used to create a signature.**

These commands instruct PGP to encrypt our message file using only ASCII characters and to use our computer's linefeed options. Using the a command is important since some Internet E-mail systems will accept only plain ASCII text characters in a message. PGP then signs the letter with the name of the sender and prompts us for our pass phrase.

At this point, PGP encrypts the file and names it ANNLTR.ASC. Now we can use our E-mail to send Ann a short message in which we will either enclose the ANNLTR.ASC file's text or attach the ANNLTR.ASC file as an enclosure to the message, depending on the desire of the sender or the strength of the E-mail program. If placing the encrypted text of the ANNLTR.ASC file in the body of the message, it might look as follows:

From: HG_Wells@novels.com
To: AnnVeronica@prison.gov

This is a recipe for a file, I mean cake, hope you're doing fine.

--BEGIN PGP MESSAGE--
Version: 2.6.2

Asdfasdfsghy tnymim,ftnyunmom nfy83w7 nruy9m e6bn890,e6bont7iym9,
8mytr5654nfghkop-8]./],[mtnd5qa2wezfvgyn8y9,o uicn80ui „--9]i,jhugiue4
e56tw6ni870„pw4be67u9iuohy8u8uedr89yu0omn87ygwervybjun89586q233
2w8=;\].,;ymytyr21qWA3WE45VKI'Λ=]DSWEDNY8IM9U07TR5N76TB
WBMKOL,HGCFYTHNUINSEDRFSRFBYUJNFGUIM,IOP;MHUYNU9
REE45RWBYNM,LOPM;UIHJBUMUO,UYTGBFTNGUIM,KIP;.[]'N
JNBHBGRT7H8UAbnygmhik
jnimgghjhbseDBFTYnYguNRUFGUNRTYGYJIMTrtbyntjENDRcftyHUt
UYhUtyIUmfTyurFuhUFhIbYdnRYNYFybtuNo<pl">? DrdF5E%NUL
UhbHuN6mL
Frtnyuyk< tfYhNM:.CvbnKMLnbbNImLGY*)Ui(FTyGJY&Y#Q$ ER
mt<P{>; =$ITY
--END PGP MESSAGE--

When Ann opens her mail and reads this message, she would then use PGP to decrypt it. First, she will have to save the enclosed file or the message as HGWLTR.ASC, and then, to decrypt the file, she'll use the PGP command;

pgp HGWLTR.ASC

The PGP program will then prompt Ann to enter her pass phrase. After correctly entering her pass phrase, PGP will let her know that we (H.G.) signed the file and will then produce a text copy of the original unencrypted ANNLTR file. She can use any text editor or word processor to read it.

If Ann has not lost her computer and network rights from the message, she can write her reply, whereupon she will use our public key to encrypt it. With PGP, we don't have to worry about anyone reading our private messages.

At this point I hope you have learned some of the basic features of PGP. I recommend that you take some time to read the documents that come with the program or one of the many books now published on the software product. They include the many commands and methods of use and can only add to the power of Phil's software. (Note: I understand version 3 will be available sometime in the near future.) With this presentation you should be able to begin sending and receiving encrypted mail or protecting that confidential data or that prize-winning manuscript.

Now for the rough part. Where can I get PGP? You can find PGP on shareware disks, bulletin board systems, and Internet sites, but the most secure and controlled place to procure PGP is from the Massachusetts Institute of Technology (MIT), the official PGP distributor for noncommercial use of the software. For commercial use of the product get in touch with ViaCrypt, they are the licensed reseller of PGP.

You can get the PGP program by way of a Web browser, FTP, or several online services. If you have full World-Wide Web access with software such as Lynx, Mosaic or Netscape, you can find PGP at MIT using the following context:

http://web.mit.edu.network/pgp-form.html

But before you can download PGP, you must read two licensing agreements online and answer the questions in an electronic form similar to the one shown below:

Are you a citizen or national of the United States or a person who has been lawfully admitted for permanent residence in the United States under the Immigration and Naturalization Act?
Yes or No

Do you agree not to export PGP 2.6.2 or RSAREF to the extent incorporated therein, in violation of the export control laws of the United States of America as implemented by the United States Department of State office of Defense Trade Controls?
Yes or No

Do you agree to the terms and conditions of the RSAREF license (in /pub/PGP/rsalicen.txt)?
Yes or No

Will you use PGP 2.6.2 solely for non-commercial purposes?
Yes or No

Submit

You also must agree not to export PGP to anyone outside the US. None of this information identifies you, I am told. MIT merely asks these questions to adhere to US laws concerning encryption technology, I am told. After you answer the questions, click the Submit button.

You will then be able to download the PGP file to your system (maybe!). Can't download PGP right away? Could be that the MIT server is busy as it allows only about 20 downloads at any one time, I am told. So simply return to the previous MIT page and submit your information again. If the server is still busy, you might want to try again after regular business hours (beats me what they are.) There are a couple of other reasons the server at MIT might not let you download PGP.

Number One: The hidden directory that holds the PGP software has changed since you started the process. The name of the directory that holds the software changes every 30 minutes (on the hour and the half hour.) Because of this, you must get the PGP files in the same half-hour period that you answer the questions. So check your time and hope that network traffic is not high. The directory context that holds the software looks something like the following:

/pub/PGP/dist/U.S.-only-xxxx (where xxxx is a randomly generated set of digits and/or letters)

Note: the UNIX , VAX and a few other versions are also located here.

Number Two: The MIT server will prevent you from downloading PGP if it thinks your host or network or Internet service provider is outside the United States. If you get a message saying that you're outside the US and you aren't, send E-mail to postmaster@net-dist.mit.edu and include the domain name of your Internet connection, such as amazon.com and try to convince them that you re not located along the river somewhere. Those of you who use a university's domain name will probably not have this problem. Those who use technet.nm.org, should not have a problem because I've already convinced them that we are not located in Mexico. And they have added it to the OK list at MIT.

Once you pass MIT's security (?) check, you ll get a list of PGP files to choose from. MS-DOS users should select PGP262.ZIP. Macintosh users will need MacPGP2.6.2sea.hqx. Also remember to get a copy of PKUNZIP to unzip the files in PGP262.ZIP if you don't have it. You can find the latest version in a file called pkzip24g.exe on most anonymous FTP servers.

For those of you that use FTP to acquire things get the following file:

ftp://net-dist.mit.edu/pub/PGP/README

As with the Web, the FTP location of PGP changes also, this is to ensure that you read the licensing and non-export agreement before you obtain PGP from MIT.

I have read that if you're using the Internet through America Online (AOL), you can get PGP by going to the AOL software library (KEYWORD: Software) and performing a search for PGP. On CompuServe, you type GO NCSAFORUM, follow the instructions to gain access to Library 12 (it is export controlled) and then look for the file PGP262.ZIP. I have yet to receive permission.

I have also read articles that state that you can get PGP via your E-mail. If you do not have access to FTP, send a message that says "help" to: ftp-request@netcom.com or mailserv@nic.funet.fi. I assume you will receive instructions on how to get PGP in uuencoded form and you will then need a uudecoding program such as WinCode or Uundo to prepare PGP for use.

For Windows users there are some public-domain and shareware programs that will interface with Pretty Good Privacy so that you will not have to launch a DOS window. One of these programs is WinPGP, an easy-to-use Windows-based PGP shell. WinPGP costs $29.00 to register and upgrades are free. To find a copy of WinPGP on the World-Wide Web or through FTP, go to either of these Internet sites.

http://www.firstnet.net/pub/windows/winpgp/pgpw31.zip
ftp://ftp.firstnet.net/pub/windows/winpgp/pgpwin31.zip

There is a version of PGP for OS/2 and it is located at:

http://www.gibbon.com/getpgp.html
ftp://ftp.gibbon.com

In closing this presentation I would like to mention the UseNet News group for PGP alt.security.pgp. It has the typical, many and varied persons of such a group but is also one of the best sources of information relative to PGP.

I would also remind system supervisors and administrators - be aware of your system's data. In addition to pretty good privacy, PGP also provides an excellent tool for users to ransom you data files. So keep your backup files current - I'm quite sure that NSA will NOT allow you to use their resources to decrypt your data.

References

Pretty Good Privacy <tm>
Documentation Files

PGP
Simson Garfinkel
O'Reilly and Associates